Development of Real Time Embedded Software for Video Compression using POWERPC

Anjaly Gopinath, Femy John and M. Vengatachalam

Abstract *--- In this paper the design method for developing an application for recording and storing time varying data displayed on the monitor based on embedded Linux using Qt and X11 is presented. A graphical user interface is designed to provide easy accessibility and flexibility to the user to store the video capture by Graphics PMC card. This finds application in post trial analysis of time varying results. The hardware platform is IBM's PPC 7410 board; while the software platform is Embedded Linux and the development environment is based on X11 and Qt/embedded. The board has MontaVista Linux embedded in it with X11 built as an abstract layer on kernel providing graphics capability to the system. VGA out of the Argus graphics card is connected to CRT monitor. Time varying data on the monitor is captured and FFmpeg library is employed to encode window frames to mpeg video which is then stored to USB. Graphics card is attached to PMC slot on the board. The experimental testing and results indicated that the system is working stably and reliably. The implementation of the entire design environment is based on the X Window system and Embedded Linux and can thus be used on an increasing number of low cost work stations.*

Keywords--- *X WIndow System, Qt, X11, Graphics PMC Card, PowerPC*

I. INTRODUCTION

THE rapid development of embedded chip technology has lead to the evolvement of embedded system as high-powered, multifunctional and all-purpose, which has had a significant impact on changing people's lifestyle and improving quality of life[1] . It has made its mark in every field ranging personal home affairs, business, process automation in industries, defence etc. Many of the applications running on embedded platforms generate time varying visualizations as results on the display device. On-line and post-trial analyses of the data need to be performed allowing the visualization of the trials measurements and comparison to predicted or modelled results. Following the trials, a period of intense data analysis occurs where the measured data from the disparate sources may be modified and compared. When the time varying results are displayed on the monitor connected to embedded board, it needs to be recorded as a video for the required amount of time so that further analysis is possible.

Anjaly Gopinath, M. Tech student, Department of ECE, Viswajyothi College of Engineering & Technology, Vazhakulam, Kerala, India, E-mail: anjaly88@yahoo.com

Femy John, Asst. Professor, Department of ECE, Viswajyothi College of Engineering & Technology, Vazhakulam, Kerala, India, E-mail: femyunni@gmail.com

M. Vengatachalam, Scientist, Naval Physical Oceanographic Laboratory, Thrikkakara, Cochin, Kerala, India, E-mail: chalam68@hotmail.com

Applications running on POWERPC based Embedded Linux system DP-VME-0504 board generate time varying results. These results are displayed on the PowerPC Monitor display. Argus graphics PMC [2] card is attached to one of the PMC slots on the PPC board. VGA out of the card is connected to the monitor display. For carrying out post trial analysis output of Graphics card is taken and converted to video so that it can be replayed when necessary. The results are captured from the VGA out of graphics PMC card , converted to video by the use of FFMPEG and can routed through USB. A Graphical User Interface is provided by Qt and C++ for flexibility.

II. SYSTEM PLATFORM STRUCTURE

The hardware platform is DP-VME-0504 PPC7410 board. The board has MontaVista Linux as the embedded operating system and X11 window system built as an abstract layer on embedded linux kernel to provide graphic capabilities for the system. PowerPC (short for *Performance Optimization With Enhanced RISC – Performance Computing*, sometimes abbreviated as PPC) is a RISC architecture created by the 1991 Apple–IBM–Motorola alliance, known as *AIM*. PPC 7410 is the latest powerpc processor (7410 with Alti Vec) @ 400 MHz with an inbuilt 2MB L2 cache. The board has up to 32 MB boot flash, 32 MB user flash and up to 128 MB SDRAM with ECC[3]. Figure 1 shows the system platform structure of the system. Argus graphics PMC card is attached to one of the PMC slots on the PPC board. VGA out of the card is connected to the monitor display. Argus graphics PMC card supports an analog (RGB) resolution up to 1920 x 1200. It has a USB interface to which a USB driver can be attached.

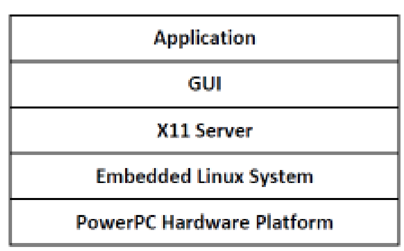


Figure 1: System Platform Structure

The X window system (commonly X Window System or X11, based on its current major version being 11) is a computer software system and network protocol that provides a basis for graphical user interfaces and rich input device capability for networked computers[6]. It creates a hardware abstraction layer where software is written to use a generalized set of commands, allowing for device independence and reuse of programs on any computer that implements X. The X protocol completely defines the capabilities of the X Window system. Figure 2 shows the architecture of X Window System. Xlib is an X Window System protocol client library written in the C programming language. Xlib allows to draw graphics on the screen of any X server, local or remote, using the C language. All needed to be done is to include <X11/Xlib.h>, link your program using the -lX11 switch, and its possible to use any of the functions in the library. It contains functions for interacting with an X server.

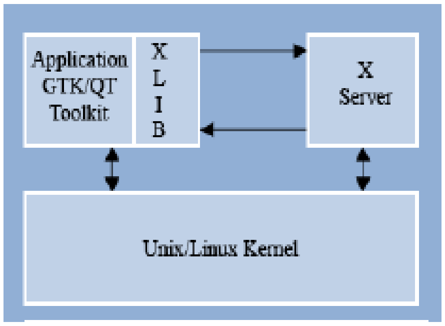


Figure 2: X Window System

Qt is the development language for application. Qt is a flagship product of Qt Development Frameworks. Qt/embedded is the C++ framework for GUI and application development on embedded devices. It runs on variety of processors with Embedded Linux. Qt is object oriented, component based and has a rich variety of widgets available at the disposal of a programmer to choose from. The most important features of Qt are signals and slots[12].

Application written in C++ using the Qt toolkit as platform abstraction layer. ***Xlib*** is used to interface with the X11 server, the X11 ***Xcomposite extension*** is used to read image data from X11 captured window and ***libav libraries*** are used to encode the data into a video file. Figure 3 shows the Qt/Embedded framework.

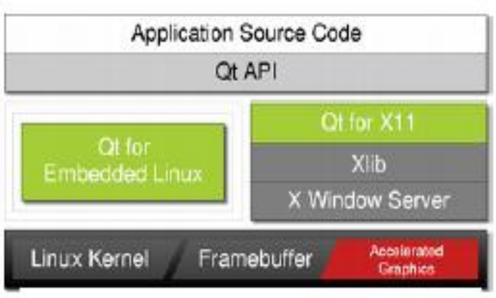


Figure 3: Qt/Embedded Framework

A. *Qt Eventhandling*

In Qt, events are objects, derived from the abstract QtEvent class that represents things that have happened either within an application or as a result of outside activity that the application needs to know about. X continually sends events to the application where they are demarshalled and inserted into an event queue. Application pulls the events out of the queue one at a time. When damage notify event received, application reads the updated image data into the shared memory segment. Thus, continuous capture done. The capture process and tracking changes to the window done by one thread. Encoding functionality is carried out separately in another thread

B. *Qt Multithreading*

Qt provides a portable API for creating and synchronizing threads since version 2.2 and offers the option of building the Qt library with or without thread support. QReadWriteLock is used to achieve synchronization between the threads. The QReadwrite Lock class provides read-write locking. This type of lock is useful if you want to allow multiple threads to have simultaneous read-only access, but as soon as one thread wants to write to the resource, all other threads must be blocked until the writing is complete.

C. *Libav libraries*

Libav libraries are a part of the video codec library Ffmpeg[13]. Ffmpeg is a complete cross-platform solution to record, convert and stream audio and video. The code is written in C. It can be compiled under most operating systems. To compress the captured window frames into mpeg, libav library routines are used. Various properties of mpeg video file like frame rate, gop size may be specified.

III. Design and Implementation

The system consists of an embedded linux system running an application based on Qt/Embedded. Figure 4 shows the layout of a typical cross-development environment[14]. A host PC is connected to a target board via one or more physical connections. It is most convenient if both serial and Ethernet ports are available on the target. Minicom is one of the most commonly used serial terminal applications and is available on virtually all desktop Linux distributions. We plug the target board's Ethernet interface into a local Ethernet hub or switch, to which the development host is also attached via Ethernet. The development host contains development tools and utilities along with target files normally obtained from an embedded Linux distribution.

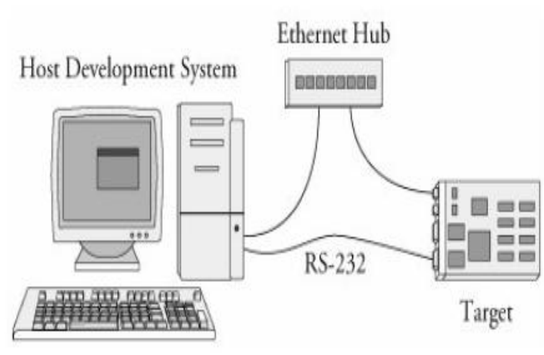


Figure 4: Cross Development Environment

Qt is built with qmake tool which can be used for building the project. qmake takes the project file and combines it with a predefined set of templates suitable for the platform to form a Makefile. The source files, headerfiles, and the paths to the libXComposite.so, libXdamage.so and the libav libraries needs to be included in the project file. Qt project is cross compiled to produce executable for the target board.

The Qt Everywhere 4.6.3 archive is used for cross compilation of the application developed on the host system. We have to build the Qt Embedded 4.6.3 libraries for Embedded Linux by editing the appropriate qmake.conf file and setting path to the cross compiler powerpc-linux-g++ which must be present in the host.

The below libraries must be copied from qt-embedded-opensource -src-4.6.3/lib/ on host to /usr/lib on target.

libQtCore.so.4

libQtGui.so.4

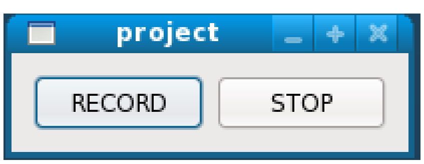
libQtNetwork.so.4

This can be done by copying the above libraries to /usr/lib of the target rootfile system. Target applications are built by means of Qt's qmake tool.

The application can then be transferred to and executed on target. Also other dependencies for the application like libavcodec.so, ibavformat.so, libswscale.so, libXComposite.so and libXDamage.so also must be copied to /usr/lib on target. The rootfile system can then be mounted on to the PPC target board via target NFS root mount. The application does exchange of image data between itself and the server using the shared memory. XShmCreateImage allocates the memory needed for the Ximage structure. The function shmget creates a new shared memory segment.XShmAttach tells server to attach to the shared memory segment. Tracking damages to the window can be done by the use of damage extension. Damage handle created for the window which reports an event when window state changes from not damaged to damaged. Ffmpeg library is used for compressing the frames captured to MPEG4 video. The parameters to be given as input for codec are frame rate , bit rate , resolution . As the frame rate decreases , size of video file decreases.

IV. RESULTS

The rootfile system is mounted on to the PPC target board via target NFS root mount. When the application is executed on the target, GUI will appear on the monitor.



Pressing the RECORD button will start capturing the video frames and it will continue till the STOP button is pressed. The MPEG file plays for an amount of time for which capturing was done. This was done by setting the frame rate to an appropriate value according to the system performance. The MPEG file that has been created has no color reverses. There is no flickering in the final MPEG file.

V. CONCLUSION

A design method of recording time varying data in an embedded platform is proposed. It is implemented in DP-VME-0504 powerpc platform and MPEG file is stored to USB drive attached to the powerpc board and can be played in any standard player. The using of Qt/Embedded greatly increases the realibility of the system and makes the codes to run in cross-platform environments. GUI provides easy flexibility to the user. Open-source Ffmpeg is used to compress the window frames to video. Experimental results show that the system fulfils the design requirements in the performance, running speed and stability.

REFERENCES

[1] Wang Zheng, Lin Xiaochuan, Zhou Yun-Lian, Ouyang Tianli. "Design and Implementation of Multi- media Player System Based on QT4 & Linux" Journal of Guizhou University (Natural Science Edition), 2009,(1):PP 60-64.

[2] Argus Graphics PMC Manual.

[3] PPC 7410 Datasheet.

[4] Karim Yaghmour.Building Embedded Linux Systems. O'Reily Media. April 2003.

[5] Embedded Linux System Design and Development – P.Raghvan.Amal Lad.Sriram Neelakandan .

[6] Eric F. Johnson and Kevin Reichard, X Window Applications Programming, Tech publications,Singapore, 1989.

[7] Michael A. Dolan Larry Hare. X Window SystemServers in Embedded Systems. 1990 IEEE.

[8] Eric F Johnson, Kevin Richard: X Window Applications Programming, M&T Books, 1990.

[9] Uday O Pabrai,Hement T Shah,: X Window System User's Guide, The Artech House Telecommunications Library , 1994

[10] Nabajyoti Barkakati: X Window System Programming, Prentice Hall Inc, Edition 2, 1996

[11] Xlib Programming Manual

[12] Trolltech. Qt Reference Documentation [EB/OL] http://doc.trolltech.com/2.3/index.html.2004.

[13] http://www.ffmpeg.org

[14] Embedded Linux Primer: A Practical Real World Approach – Christopher Hallinan .

[15] C. Schofield, M. S. Ashley. Advanced data storage & analysis for oceanographic trials . Electronic Engineering in Oceanography, 19-21 July 1994, Conference Publication No 394.

Anjaly Gopinath was born in India. She took her B-Tech in Electronics and Communication Engineering from Adi Shankara Institute of Engineering And Technology, Kalady (2010) and is presently doing her M-Tech in VLSI and Embedded System from Viswajyothi College of Engineering and Technology, Muvattupuzha. Her fields of interest include Digital Signal Processing and Wireless Communication

Femy John was born in India. She took her B-Tech in Electronics and Communication from Viswajyothi College of Engineering and Technology, Muvattupuzha (2007) and M-Tech from Model Engineering College, Thrikkakara. She is currently working as Assistant Professor in Viswajyothi College of Engineering and Technology, Muvattupuzha.

M Vengatachalam was born in India. He is currently working as Scientist in Naval Physical and Oceanographic Laboratory (NPOL), an underwater surveillance laboratory of the DRDO.